DTU Library

Entropy of Bit-Stuffing-Induced Measures for Two-Dimensional Checkerboard Constraints

Forchhammer, Søren; Vaarby, Torben Strange

Published in:
I E E E Transactions on Information Theory

Link to article, DOI:
10.1109/TIT.2007.892781

Publication date:
2007

Document Version
Publisher's PDF, also known as Version of record

$|\mathcal{D}|/2 = n2^{k-1}$. Similarly, we get $|\mathcal{D}_2| = n2^{k-1}$. The claim (**) is proved.

By the assumption of induction $\mathcal{D}_1$ contains an element $\bar{u}'$ from $(Z_{2^k}^*)^{n/2}$. This means that there exists $\bar{u}''$ from $Z_{2^k}^{n/2}$ such that $(\bar{u}', \bar{u}'') \in \mathcal{D}$. Since $wt^*(2^{k-1}(\bar{u}', \bar{u}'')) \in \{n2^{k-2}, n2^{k-1}\}$ and $2^{k-1}\bar{u}' = (2^{k-1}, \ldots, 2^{k-1})$, we have $wt^*(2^{k-1}\bar{u}'') = 0$ or $wt^*(2^{k-1}\bar{u}'') = n2^{k-2}$. So

$$2^{k-1}\bar{u}'' = (0, \ldots, 0) \tag{1}$$

or

$$2^{k-1}\bar{u}'' = (2^{k-1}, \ldots, 2^{k-1}). \tag{2}$$

Similarly, considering the code $\mathcal{D}_2$, we can find a codeword $(\bar{v}', \bar{v}'') \in \mathcal{D}$ such that $2^{k-1}\bar{v}'' = (2^{k-1}, \ldots, 2^{k-1})$ and v' satisfies

$$2^{k-1}\bar{v}' = (0, \ldots, 0) \tag{3}$$

or

$$2^{k-1}\bar{v}' = (2^{k-1}, \ldots, 2^{k-1}). \tag{4}$$

If (2) is true, then $(u', u'') \in (Z_{2^k}^*)^n \cap \mathcal{D}$. If (4) is true, then $(v', v'') \in (Z_{2^k}^*)^n \cap \mathcal{D}$. If (1) is true and (4) is true, then $(u', u'') + (v', v'') \in (Z_{2^k}^*)^n \cap \mathcal{D}$. Lemma 6-4 is proved. □

Proof of Theorem 6-3: By Lemma 6-4 the code $\mathcal{D}$ contains an element $\bar{c} = (c_1, \ldots, c_n)$ from $(Z_{2^k}^*)^n$. Then the code $\mathcal{D}' \triangleq (c_1^{-1}, \ldots, c_n^{-1}) \circ \mathcal{D}$ is equivalent to $\mathcal{D}$ and contains $\bar{1} = (1, \ldots, 1)$. Let $\{\bar{1}, q_1, \ldots, q_s\}$ be a basis of $\mathcal{D}'$ and i_j be the number of elements of order $2^j, j = 1, \cdots, k$. Assume $\bar{1}, q_1, \ldots, q_s$ are the rows of the matrix Q of size $(1 + i_1 + \cdots + i_k) \times n$, a generator matrix of the code $\mathcal{D}'$. Then the columns of Q are elements of $\{1\} \times (2^{k-1}Z_{2^k})^{i_1} \times (2^{k-2}Z_{2^k})^{i_2} \times \ldots \times (2^0 Z_{2^k})^{i_k}$. Since by Lemma 5-1 the code $d^\diamond$-distance of $\mathcal{D}'^\perp$ is more than 2, all the columns are pairwise different. Therefore Q coincides with $B_I, I = (i_1, \ldots, i_k)$, up to permutation of columns. □

So, we conclude, provided the mappings φ and Φ are fixed, all up to equivalence co-Z_{2^k}-linear extended 1-perfect codes and Z_{2^k}-linear Hadamard codes are described in Sections IV and V.

REFERENCES

[1] A. A. Nechaev, "Trace-function in Galois ring and noise-stable codes," in *Proc. V All-Union Symp. Theory of Rings, Alg. Mod.*, Novosibirsk, Russia, 1982, p. 97, in Russian.

[2] ——, "Kerdock code in a cyclic form," *Discr. Math. Applicat.*, vol. 1, no. 4, pp. 365–384, 1991.

[3] A. R. J. Hammons, P. V. Kumar, A. R. Calderbank, N. J. A. Sloane, and P. Solé, "The Z_4-linearity of Kerdock, Preparata, Goethals, and related codes," *IEEE Trans. Inf. Theory*, vol. 40, no. 2, pp. 301–319, 1994.

[4] C. Carlet, "Z_{2^k}-linear codes," *IEEE Trans. Inf. Theory*, vol. 44, no. 4, pp. 1543–1547, 1998.

[5] T. Honold and A. A. Nechaev, "Fully weighted modules and representations of codes," *Probl. Inf. Transm.*, vol. 35, no. 3, pp. 205–223, 1999.

[6] I. Constantinescu and W. Heise, "A metric for codes over residue class rings of integers," *Probl. Inf. Transm.*, vol. 33, no. 3, pp. 208–213, 1997.

[7] V. A. Zinoviev, "Generalized concatenated codes," *Probl. Inf. Transm.*, vol. 12, no. 1, pp. 2–9, 1976.

[8] J. Rifa and J. Pujol, "Translation-invariant propelinear codes," *IEEE Trans. Inf. Theory*, vol. 43, no. 2, pp. 590–598, 1997.

[9] J. Borges and J. Rifa, "A characterization of 1-perfect additive codes," *IEEE Trans. Inf. Theory*, vol. 45, no. 8, pp. 1688–1697, 1999.

[10] D. S. Krotov, "Z_4-linear perfect codes," *Diskr. Analiz i Issled. Operatsii, Ser.1*, vol. 7, no. 4, pp. 78–90, 2000, in Russian.

[11] ——, "Z_4-linear Hadamard and extended perfect codes," *Electron. Notes in Discr. Math.*, vol. 6, pp. 107–112, 2001.

[12] K. T. Phelps and J. Rifa, "On binary 1-perfect additive codes: Some structural properties," *IEEE Trans. Inf. Theory*, vol. 48, no. 9, pp. 2587–2592, 2002.

[13] J. Borges, K. T. Phelps, and J. Rifa, "The rank and kernel of extended 1-perfect Z_4-linear and additive non-Z_4-linear codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 8, pp. 2028–2034, 2003.

[14] K. T. Phelps, J. Rifa, and M. Villanueva, "Rank and kernel of additive (Z_4-linear and non-Z_4-linear) Hadamard codes," in *Proc. 9th Int. Workshop on Algebraic and Combinatorial Coding Theory ACCT'2004*, Kranevo, Bulgaria, Jun. 2004, pp. 327–332.

[15] K. T. Phelps, J. Rifa, and M. Villanueva, "On the additive (Z_4-linear and non-Z_4-linear) Hadamard codes. Rank and kernel," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 316–319, Jan. 2006.

[16] M. K. Gupta, M. C. Bhandari, and A. K. Lal, "On linear codes over Z_{2^s}," *Designs, Codes Cryptogr.*, vol. 36, no. 3, pp. 227–244, 2005.

[17] F. I. Solov'eva, "On the construction of transitive codes," *Probl. Inf. Transm.*, vol. 41, no. 3, pp. 204–211, 2005.

[18] F. J. MacWilliams and N. J. A. Sloan, *The Theory of Error-Correcting Codes*. Amsterdam, Netherlands: North Holland, 1977.

[19] R. Ahlswede, H. K. Aydinian, and L. K. Khachatrian, "On perfect codes and related concepts," *Designs, Codes Cryptogr.*, vol. 22, no. 3, pp. 221–237, 2001.

[20] *Translated from Diskretnaya Matematika*, vol. 1, no. 4, pp. 123–139, 1989.

[21] *Translated from Probl. Peredachi Inf.*, vol. 12, no. 1, pp. 5–15, 1976.

Entropy of Bit-Stuffing-Induced Measures for Two-Dimensional Checkerboard Constraints

Søren Forchhammer, *Member, IEEE*, and Torben V. Laursen

***Abstract*—A modified bit-stuffing scheme for two-dimensional (2-D) checkerboard constraints is introduced. The entropy of the scheme is determined based on a probability measure defined by the modified bit-stuffing. Entropy results of the scheme are given for 2-D constraints on a binary alphabet. The constraints considered are 2-D RLL (d, ∞) for $d = 2, 3$ and 4 as well as for the constraint with a minimum 1-norm distance of 3 between 1s. For these results the entropy is within 1–2% of an upper bound on the capacity for the constraint. As a variation of the scheme, periodic merging arrays are also considered.**

***Index Terms*—Bit-stuffing encoding, cascading two-dimensional (2-D) arrays, run-length-limited (RLL) constraints, 2-D constraints.**

I. INTRODUCTION

Constrained coding has found widespread use in optical and magnetic data storage devices. Writing data along tracks, constrained codes [1] have been applied to increase the density for a given physical media and facilitate synchronization. Codes have been designed, e.g., for *run-length-limited* RLL(d, k) constraints, where the number of 0s between successive 1s has to be at least d and at most k.

New developments in storage medias such as holographic storage and advances in nano storage such as the millipede project [2], [3] are possible application areas for two-dimensional (2-D) constrained codes and fields. An advantage of 2-D constrained coding could be to increase

Manuscript received November 28, 2005; revised December 22, 2006. The material in this correspondence was presented in part at the IEEE International Symposium on Information Theory, Chicago, IL, June/July 2004.

S. Forchhammer is with the Research Center COM, B.343, Technical University of Denmark, DK-2800 Lyngy, Denmark (e-mail: sf@com.dtu.dk).

T. V. Laursen was with the Research Center COM, B.343, Technical University of Denmark, DK-2800 Lyngy, Denmark. He is now with the TrygVesta, DK-2750 Ballerup, Denmark (e-mail: vaarby@gmail.com).

Communicated by G. Zémor, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2007.892781

$$\mathcal{N}_{\diamond(3)}: \begin{matrix} & & 0 & & \\ & 0 & 0 & 0 & \\ 0 & 0 & 1 & 0 & 0 \\ & 0 & 0 & 0 & \\ & & 0 & & \end{matrix} \qquad \mathcal{N}_{2,\infty}: \begin{matrix} & & 0 & & \\ & & 0 & & \\ 0 & 0 & 1 & 0 & 0 \\ & & 0 & & \\ & & 0 & & \end{matrix}.$$

Fig. 1. The 0-neighborhood, $\mathcal{N}$ for the constraints $\diamond(3)$ and 2-D RLL$(2, \infty)$.

the density. For example 2-D RLL constrained coding has been considered [4].

The constrained 2-D fields to be considered are specified by shift invariant constraints of finite extent (N, M). A constraint is defined by a list, $\mathcal{F}$, of forbidden blocks of maximum size $N \times M$ made of symbols from a finite alphabet A. A configuration on an n by m rectangle having no forbidden blocks within the rectangle is called an admissible configuration. This corresponds to a $\mathcal{Z}^2$ shift of finite type [5], [6].

Let $E(n, m)$ be the set of admissible configurations on an n by m rectangle for a given field F. The *capacity* or combinatorial entropy of F is defined as

$$H^{(2)}(F) = \lim_{n,m\to\infty} \frac{\log_2 |E(n,m)|}{nm}. \quad (1)$$

The limit is indeed well defined [5], [7].

This correspondence shall focus on the 2-D RLL(d, ∞) constraint, which is an example of a checkerboard constraint [8]. Bit-stuffing may be applied to checkerboard constraints. Bit-stuffing for 2-D constraints as in [4], [11] is modified in a way which enables analysis based on a probability measure induced by the scheme. The entropy of the measures may be calculated, which besides the capacities of specific bit-stuffing schemes provides a lower bound on the capacity of the constraints considered. In [4], a detailed study of the relation of actual bit-stuffing coding schemes and the measure induced was presented. Here we restrict the analysis to determining the entropy of the scheme based on a measure.

Section II presents checkerboard constraints and bit-stuffing. The modified bit-stuffing scheme and the related probability measure are presented in Section III. Based on the measure, the entropy of the scheme is given. Section IV introduces periodic merging arrays, which may be seen as a special case of the measures used for bit-stuffing. Numerical results for the entropies are given in Section V.

II. Bit-Stuffing for Checkerboard Constraints

A. Checkerboard Constraints

Consider binary constraints where each 1 has to be surrounded by an arbitrary, but specific neighborhood of 0s, referred to as a 0-neighborhood, $\mathcal{N}$. These *checkerboard constraints* were defined and treated in [8] and further investigated in [9]. In the latter article the neighborhood could be any measurable, bounded subset of the plane, but we will restrict ourselves to neighborhoods defined on the integer lattice $\mathcal{Z}^2$. The RLL(d, ∞) constraint is an example of this. Another example is the diamond constraint or minimum distance M between ones with regards to the 1-norm [10]. We will refer to this constraint as $\diamond(M)$ (in [8] this constraint is referred to as the Diamond $M - 1$ constraint).

The extent of the $\diamond(M)$ constraint is $M \times M$. However, unlike the 2-D RLL(d, k) constraint, the forbidden words do not split along rows and columns. The 0-neighborhood, $\mathcal{N}$, required around a 1 for each of the two constraints $\diamond(3)$ and 2-D RLL$(2, \infty)$ is depicted in Fig. 1.

Given a 0-neighborhood, $\mathcal{N}$, specifying a checkerboard constraint, the extent $(N \times M)$ of the checkerboard constraint is given by the smallest values of N and M, for which the $(2N-1)$ by $(2M-1)$ rectangle centered at the 1 ontains the 0-neighborhood.

B. Bit-Stuffing

Bit-stuffing is a simple, yet efficient way to code for 1-D RLL constraints. The method has been extended to 2-D constrained arrays, [4], [11], [12]. It is applicable if it is always possible to write say a 0 at any position. The elements of the 2-D array is traversed in a predefined order, e.g., row by row or along diagonals. The data stream may be written as is, except that each time a 1 is encountered, the necessary number of zeros are stuffed immediately after. Constraints of this type include 2-D RLL(d, ∞) and other checkerboard constraints [8]. An analysis of 2-D bit-stuffing has been presented [4], showing that it is very efficient for the 2-D RLL$(1, \infty)$ constraint. In [11], bit-stuffing was used and analyzed for the 2-D RLL(d, ∞) constraint. Analysis of the hard-triangle has also been carried out [12]. In [4], [11] the bit-stuffing is performed along diagonals, writing bits from a sequence whenever possible and writing the 0's the constraint prescribes. The iid unbiased data sequences to be coded may be transformed into iid biased sequences in a precoding step in order to increase the entropy. One can extend this by having more than one biased sequence and choose between these depending on past data besides what is prescribed by the constraint [4].

For the hard-square [4] and hard-triangle [12], the entropy of the bit-stuffing scheme has been determined and optimized. For the higher order constraints 2-D RLL(d, ∞), $d > 1$, lower bounds on the entropy of the bit-stuffing scheme are presented in [11]. We shall modify the traversal of the elements of the 2-D code. This enables the derivation of an expression of the entropy, which in turn provides an improvement of the bounds for $d = 2, 3$, and 4, i.e., for the higher order constraints (see Section V).

C. Finite State Sources

The analysis of the new bit-stuffing scheme is based on results for 1-D sequences. In one dimension, sequences satisfying a constraint on N consecutive symbols such as run-length-limited sequences may be described by finite state sources, where a state is characterized by $N - 1$ symbols. The 1-D entropy is then defined as in (1) but with $m = M = 1$ and $n \to \infty$. The transfer (or adjacency) matrix $\mathbf{T}$ of the source indicates the possible transitions between two states. The largest eigenvalue Λ of the transfer matrix $\mathbf{T}$ determines the growth rate of the number of configurations. Taking the logarithm gives the maximum entropy [13]

$$H^{(1)} = \log_2(\Lambda). \quad (2)$$

The one-dimensional approach is readily generalized to 2-D arrays of finite (horizontal) width m and arbitrary (vertical) height n. The admissible configurations of an array of width m may for all n be described by a finite state source. For a constraint of extent (N, M), the states of the source are given by the symbols on the $(N-1) \times m$ segment which appear as the first or last $N - 1$ rows of an admissible configuration on a $N \times m$ rectangle, i.e., a configuration of $E(N, m)$. A transition from state i to state j is admissible if there is a configuration in $E(N, m)$, for which state i is identical to the top $N - 1$ rows and state j to the bottom $N - 1$ rows. State i and j have an overlap of $N - 2$ rows. The last row of j is generated by the transition from i to j and appended to the previous rows of the output. Any admissible configuration of $E(n, m)$ with fixed m and $n (> N - 1)$ rows may be generated as an output by starting the source in the state specified by the first $N - 1$ rows and making $n - N + 1$ transitions appending one row to the output in each transition. The transfer matrix $\mathbf{T}_m$ indicates transitions which satisfy the constraint by defining the elements $t_{ij} = 1$ if the transition from state i to j is admissible and $t_{ij} = 0$ if it is not admissible. (In the next section, a probability p_{ij} shall be assigned to the transition from i to j.) The per symbol entropy of the source on an array of width $m (n \to \infty)$ is given by

$$\frac{H(m)}{m} = \frac{\log_2(\Lambda_m)}{m} \tag{3}$$

where Λ_m is the largest (positive) eigenvalue of $\mathbf{T}_m$. Equation (3) provides an upper bound on the entropy $H^{(2)}$ defined by (1) [10].

For constraints where any two configurations, $\mathbf{x}$ and $\mathbf{y}$, on arrays of width m may admissibly be concatenated (or cascaded) by in between padding a merging array, $\mathbf{v}$, having c columns to form the admissible configuration $\mathbf{xvy}$, a lower bound on the capacity based on (3) is given by

$$H^{(2)} \geq \frac{H(m)}{m+c}. \tag{4}$$

III. Measures on Configurations

Let W denote an $n \times m$ array over some alphabet A. Let μ_W be a probability measure defined on W, that is $0 \leq \mu_W(\mathbf{x}) \leq 1$ for $\mathbf{x} \in A^{n \times m}$. In the general case, the variables $\mathbf{x}$ can take on any of the $|A|^{n \times m}$ possible values. However, we will restrict ourselves to measures agreeing with a constraint defined on A. That is, forbidden configurations are assigned probability zero.

The *entropy* of μ_W is defined as follows:

$$H(\mu_W) = -\frac{1}{nm} \sum_{\mathbf{x} \in A^{n \times m}} \mu_W(\mathbf{x}) \log_2 \mu_W(\mathbf{x}).$$

Let $b \geq M - 1$. Given an $n \times m$ rectangle W, let X and Z denote the first and last b columns, respectively, i.e., they are defined on $n \times b$ arrays. Let Y denote the middle $m - 2b$ columns. Let $XYZ \ (= W)$ denote the concatenation.

Given a probability measure, μ_W on W, consider the restriction that the (marginal) measures on the borders X and Z are identical, i.e.,

$$\mu_X \equiv \mu_Z. \tag{5}$$

Introduce the rectangles X_i, Y_i, Z_i and $W_i = X_i Y_i Z_i$, for which all X_i and Z_i have the same size as X and all W_i have the size of W. Assume $\mu_{W_i} = \mu_W$, i.e., the measures are identical, and further assume they satisfy (5). Starting with $X_0 Y_0 Z_0$, arrays $Y_i Z_i$ may repeatedly be concatenated to form the extended array

$$W_0^K = X_0 \{Y_j Z_j\}_{j=0}^K = X_0 Y_0 Z_0 Y_1 Z_1 \cdots Y_K Z_K \tag{6}$$

such that all μ_{W_i} on $W_i = Z_{i-1} Y_i Z_i$ have the same measure, μ_W. The notion to be explored is that the conditional entropy of the configurations on Y_i, Z_i conditioned on the configuration on Z_{i-1} determines the entropy of the extended array.

Defining probability measures for constrained coding in general is a challenge. One challenge, in the construction outlined above, is to ensure that the probability measures on the borders X and Z are identical, i.e., $\mu_X = \mu_Z$, satisfying (5).

Let $\mathbf{X}$ and $\mathbf{Z}$ denote stochastic variables on X and Z, respectively. A simplification is obtained by having independent borders, i.e., for all configurations $\mathbf{x}, \mathbf{z} \in A^{n \times b}$

$$P(\mathbf{X} = \mathbf{x}, \mathbf{Z} = \mathbf{z}) = \mu_X(\mathbf{x}) \mu_Z(\mathbf{z}). \tag{7}$$

Based on a finite state source with states of height $N - 1$ and width m, $\mathbf{W}$ shall be described by a finite state Markov process by assigning a probability, p_{ij}, to each transition (from state i to state j) and a probability to the initial state. Let $\mathbf{P}_m = (p_{ij})$ be the transition probabilities for the finite state Markov process and let $\mathbf{T} = (t_{ij})$ denote the corresponding transfer matrix. Throughout the paper, we require that $p_{ij} > 0$ if and only if $t_{ij} > 0$.

In case a stationary distribution π exists, $\pi \mathbf{P}_m = \pi$, the entropy per row of the Markov process $\mathbf{W}$ is defined as

$$H_W = \sum_i \sum_j \pi_i p_{ij} \log_2(1/p_{ij}), \tag{8}$$

where π_i is the stationary probability of state i given by π.

Lemma 3.1: For a Markov process $\mathbf{W}$ defined over a checkerboard constraint, the entropy, H_W given by (8) is well defined.

Proof: If a stochastic matrix is irreducible, then the stationary distribution exists and is unique. Since the stochastic matrix, $\mathbf{P}_m$ for $\mathbf{W}$ is defined such that $p_{ij} > 0$ if $t_{ij} > 0$, it is sufficient to show that the transfer matrix for the finite state description of $\mathbf{W}$ is irreducible. Let s and t be two states corresponding to any two valid configurations. For a checkerboard constraint the all zero state $\mathbf{0}$ is always valid. Consider the configuration obtained by stacking $s, \mathbf{0}$, and t on top of each other. Since this is a valid configuration, it is possible to go from state s to state t in $2N - 2$ transitions. Hence the transfer matrix is irreducible.

Actually, for $k = 2N - 2$, the matrix $\mathbf{T}_m^k$ is strictly positive, i.e., the transfer matrix is *primitive*. ∎

Note that the combinatorial entropy on W for $n \to \infty$ is given by the max-entropic solution to (8) based on the transfer matrix, $\mathbf{T}_m$ [13].

In the following, bit-stuffing schemes are used to determine the probability measure. However, the conventional bit-stuffing scheme is modified slightly in order to satisfy the requirement of independence of the borders (7).

A. Bit-Stuffing for 2-D RLL(d, ∞) Constraints

This subsection introduces the new modified bit-stuffing scheme in a basic version applicable to the 2-D RLL(d, ∞) constraint (and other checkerboard constraints where the 0-neighborhood is confined to the same row and column as the 1 symbol). First the case is considered where one or more biased sequences are used, possibly one for each column in X and Y. The main modification compared with previous bit-stuffing schemes is that the order in which the elements are addressed is not entirely contiguous within a row. Using biased sequences in bit-stuffing may be represented by conditional probabilities. In Section III-B, the scheme is presented in a more general form applicable to all checkerboard constraints.

The analysis of the modified bit-stuffing scheme is based on a description of the array as being generated by a finite-state Markov process, $\mathbf{W}$. For RLL(d, ∞) the states are $(N - 1) \times m$ symbols where $m \geq 3d$. A new element is generated in each of the m columns. Let r denote the m elements generated by a transition. The process $\mathbf{W}$ has the marginal processes at the borders, $\mathbf{X}$ and $\mathbf{Z}$, having a width $b \geq M - 1$. For RLL(d, ∞) set $b = d = M - 1$. The m new elements generated by the transition from one state, i, to the next state, j, is given by

$$\begin{aligned} r &= (r_0 r_1 \ldots r_{m-1}) \\ &= (x_0, \ldots, x_{d-1}, y_0, \ldots, y_{m-2d-1}, z_0, \ldots, z_{d-1}). \end{aligned}$$

For 2-D RLL(d, ∞), r is the new row on W. To satisfy the independence of the borders (7), the ordering of the bit-stuffing is altered slightly as

$$\tilde{r} = (x_0, \ldots, x_{d-1}, z_0, \ldots, z_{d-1}, y_0, \ldots, y_{m-2d-1}). \tag{9}$$

Let R_l denote a binary stochastic variable defined on the element in column l of the new row, r. A context variable, $C(l)$, in column l is assigned a value, $c(l)$ defined by a mapping of elements of the previous state i and elements of the new state j causal by the ordering given by $\tilde{r}$, i.e., preceding the current element in (9). The transition probability, p_{ij} of $\mathbf{P}_m$, is determined by the product of the conditional probabilities, $p(R_l = r_l | C(l) = c(l))$,

$$p_{ij} = \prod_{l=0}^{m-1} p(R_l = r_l | C(l) = c(l)) \tag{10}$$

where the same context mapping and conditional probabilities for x_k of $\mathbf{X}$ and z_k of $\mathbf{Z}$ are used, ensuring that (5) is satisfied. Let $\mathbf{P}_d$ denote the transition probability matrix of $\mathbf{X}$ obtained by only considering the bit-stuffing within X, for RLL(d, ∞).

The modified bit-stuffing scheme on the array W of width m is defined by addressing the elements in a predefined order. For the 2-D RLL(d, ∞) constraint we choose the elements row by row in the order given by $\tilde{r}$ (9). The conditional probabilities are given by (11) (shown at the bottom of the page), for $0 \leq l < m - d$. For $m - d \leq l < m$, $p_1(l) = p_1(l - m + d)$. $p(R_l = 0 | c(l)) = 1 - p(R_l = 1 | c(l))$.

Example 3.2: For the 2-D RLL$(2, \infty)$ constraint and a given width $m \geq 6$, set the width of X and Z to $b = d = 2$. The first two rows of a rectangle, W, is initialized by some admissible configuration, thereafter the elements are addressed row by row. Within each row of W the order $(\tilde{r})$ is $x_0, x_1, z_0, z_1, y_0, \ldots, y_{m-5}$, i.e., the elements of Z are addressed prior to the elements of Y within the row. With this reordering the causal elements of the 0-neighborhood of x_0 and x_1 (z_0 and z_1) are all in $X(Z)$, namely the two elements above in the same column and for $x_1(z_1)$ also the element to its right. Thus the bit-stuffing of X and Z is identical. This bit-stuffing of W may be extended by for each row repeating the reordering given by $z_0, z_1, y_0, \ldots, y_{m-5}$. Each new Z_i may be seen as X_{i+1} in relation to Y_{i+1}. The bit-stuffing of all Z_i are independent of all Y_i at the time the elements are assigned a value.

As in the example above, the bit-stuffing defined on W is extended to a larger rectangle. Now let the height of the rectangle be $n + N - 1$, such that n refers to the number of transitions. For a given d and m, let $B_{n,K}$ denote the extension of W to the array W_0^{K-1} (6), i.e., $(n + N - 1)$ rows by $(K(m - d) + d)$ columns, $0 \leq l < K(m - d) + d$.

Definition 3.3: Given $m, p_1(l), 0 \leq l < m - d$ and the first d rows of the rectangle $B_{n,K}$, the elements of $B_{n,K}$ are addressed row by row. The *modified bit-stuffing scheme* on $B_{n,K}$ for the 2-D RLL(d, ∞) constraint is defined by for each row addressing the elements of W_i in order of increasing i and within each W_i in the order given by $\tilde{r}$ (9). The same conditional probabilities (12) are applied for the elements within each W_i, i.e.,

$$p_1(l) = p_1(l \text{ modulo } (m - d)), \qquad 0 \leq l < K(m - d) + d. \tag{12}$$

When restricting the constraints to be 2-D RLL(d, ∞), the contexts, $C(l)$, are simple. Each state s consists of d rows $s_1, \ldots, s_d$. Whether it is possible to write a 1 in a given position at the time of writing is only dependent on the d previous elements in the same column and the previous elements of the current row after reordering.

The context of r_l is depicted in Fig. 2(a) for r_l in the left part of Y (meaning more than d elements away from the right border). In the right part of Y (meaning not more than d elements away from the right border) the context of r_l includes elements in both Y and Z (Fig. 2(b)) due to the reordering.

$$\begin{array}{ccccc} & s_1(l) & & s_1(l) & \\ & \vdots & & \vdots & \\ & s_d(l) & & s_d(l) & \\ r_{l-d} \cdots r_{l-1} & & r_{l-d} \cdots r_{l-1} & & z_1 \ldots z_d \\ & (a) & & (b) & \end{array}$$

Fig. 2. Examples of contexts for 2-D RLL(d, ∞). (a) shows the context in the left part of Y. (b) shows the context at position $l = m - d - 1$ (the last position in Y).

With this set of contexts, the transition probabilities, p_{ij}, (10) of $\mathbf{W}$ for the modified bit-stuffing for the 2-D RLL(d, ∞) constraint can be written as

$$\prod_{l=0}^{d-1} p(x_l | c(l)) \prod_{k=0}^{d-1} p(z_k | c(m - d + k)) \prod_{k=0}^{m-2d-1} p(y_k | c(d + k)) \tag{13}$$

where the conditional probabilities are given by (12) and $p_1(l), 0 \leq l < m - d$.

Lemma 3.4: If the bit-stuffing probabilities $p_1(l)$ satisfy $0 < p_1(l) < 1$ for all $l = 0, \ldots m - 1$, then the probabilities defined by (10)–(12) constitute a stochastic matrix, $\mathbf{P}_m$, with $p_{ij} > 0$ if and only if $t_{ij} > 0$.

Proof: The first step is to show that $p_{ij} = 0 \Leftrightarrow t_{ij} = 0$. Assume $p_{ij} = 0$ for some transition with $t_{ij} > 0$. $p(R_l = 0 | C(l) = c(l)) > 0$ for all l. Therefore, $p(R_l = 1 | C(l_0) = c(l_0)) = 0$ for some specific l_0. Hence, a 1 in position l_0 is not admissible given the context $c(l_0)$. Thus the configuration, corresponding to the transition from state i to j, is not admissible. Hence $t_{ij} = 0$. Conversely if $t_{ij} = 0$ there is some position l_0 where a 1 is not admissible but actually occur and hence $p_{ij} = 0$.

The second step is to show that (10)–(12) forms a stochastic matrix, i.e., $\sum_j p_{ij} = 1$ for all i. Consider any given i. The new elements generated by the transition to j are considered one element at a time. All admissible configurations may be represented by a complete binary tree as follows. Each time a decision may be made, i.e., $p(R_l = 1 | C(l) = c(l)) > 0$, two branches with two new nodes are created and for each new node the bit-stuffing scheme is continued. Each node may be assigned a probability namely the probability given by the product of probabilities in the path of the node. As the sum of the branching probabilities always sums to one, so will the sum over the leaves of the tree given the sum over all admissible configurations as determined by the bit-stuffing scheme. ■

Combining Lemmas 3.1 and 3.4 shows that the modified bit-stuffing scheme (11), (12) for the RLL(d, ∞) constraint leads to a Markov process $\mathbf{W}$ having a well-defined entropy given by (8). The lemmas also apply to conventional bit-stuffing without reordering, e.g., for $\mathbf{X}$ and $\mathbf{Y}$. Based on $\mathbf{W}$, a measure on the rectangle $B_{n,K}$ shall be introduced.

Initializing the Markov process $\mathbf{W}$, which is based on bit-stuffing, specifies a probability measure $\mu_{\mathbf{W}}$ on W. Based on this measure, a measure on the extension W_0^{K-1} (6), i.e., $B_{n,K}$ is specified. This measure is described in detail below leading to a family of measures,

$$p(R_l = 1 | C(l) = c(l)) = \begin{cases} p_1(l), & 0 < p_1(l) < 1, \text{ if a 1 is admissible in position } l \\ 0, & \text{if a 1 is not admissible in position } l \end{cases} \tag{11}$$

$\mathbf{x}_0$	$\mathbf{y}_0$	$\mathbf{z}_0$	$\mathbf{y}_1$	$\mathbf{z}_1$	$\cdots$	$\mathbf{y}_{K-2}$	$\mathbf{z}_{K-2}$	$\mathbf{y}_{K-1}$	$\mathbf{z}_{K-1}$
$\mathbf{x}_1$	$(\mathbf{y},\mathbf{z})_1$				$\cdots$			$(\mathbf{y},\mathbf{z})_K$	
$\mathbf{x}_2$	$(\mathbf{y},\mathbf{z})_{K+1}$				$\cdots$			$(\mathbf{y},\mathbf{z})_{2K}$	
$\vdots$	$\vdots$				$\vdots$			$\vdots$	
$\mathbf{x}_n$	$(\mathbf{y},\mathbf{z})_{(n-1)K+1}$				$\cdots$			$(\mathbf{y},\mathbf{z})_{nK}$	

Fig. 3. The rectangle $B_{n,K}$ on which the measure $\mu_{B_{n,K}}$ is defined.

$\mu_{B_{n,K}}$, for positive values of n and K. The ordering given by $\tilde{r}$ (9) assures that $\mathbf{X}$ and $\mathbf{Z}$ may be described independently (7).

Let $\mathbf{w}_0^{n,K-1}$ denote a configuration on $B_{n,K}$, composed as follows. First consider the border process $\mathbf{X}$ of $\mathbf{W}$. The transition matrix for $\mathbf{X}$ has a stationary distribution π_X by Lemmas 3.1 and 3.4. Draw the initial $(N-1)\times d$ element state $\mathbf{x}_0$ from π_X and generate n rows using $\mathbf{X}$. This is the $(n+N-1)\times d$ left hand vertical boundary $B_n^{(v)}$ of $B_{n,K}$. Let $\mu^v(\mathbf{x}_1,\dots\mathbf{x}_n|\mathbf{x}_0)$ denote the conditional measure on $B_n^{(v)}$ conditioned on the initial state $\mathbf{x}_0$.

Given the stationary distribution, π_W, of the process $\mathbf{W}$, consider a configuration $\mathbf{w} = \mathbf{x},\mathbf{y},\mathbf{z}$ and let $\pi_{\mathbf{y},\mathbf{z}|\mathbf{x}}$ be the conditional distribution $\pi_{\mathbf{y},\mathbf{z}|\mathbf{x}} = \frac{\pi_W(\mathbf{x},\mathbf{y},\mathbf{z})}{\pi_X(\mathbf{x})}$. Draw $(\mathbf{y}_0,\mathbf{z}_0)$ using this measure conditioned on the initial state $\mathbf{x}_0$ and continue to draw $(\mathbf{y}_i,\mathbf{z}_i)$ conditioned on $\mathbf{z}_{i-1}, i = 1,\dots K-1$. This constitutes the $(N-1)\times K(m-d)+d$ upper horizontal boundary, $B_K^{(h)}$, of $B_{n,K}$. Let $\mu^h(\mathbf{y}_0,\mathbf{z}_0,\dots,\mathbf{y}_{K-1},\mathbf{z}_{K-1}|\mathbf{x}_0)$ denote the conditional measure on $B_K^{(h)}$ conditioned on the initial state $\mathbf{x}_0$. The interior set of elements, $B_{n,K}\backslash(B_n^{(v)}\cup B_K^{(h)})$, is denoted $B_{n,K}^*$.

Considering the set of elements $(\mathbf{y},\mathbf{z})$, i.e., the elements of a state on Y and Z, as a single symbol of an extended alphabet, the interior $B_{n,K}^*$ may be viewed as an $n\times K$ rectangle over this alphabet (Fig. 3). For a given transition from state i to j with the combined configuration s_1^{d+1}, $\mathbf{w}_i = s_1^d$ is denoted the predecessor state of the state $\mathbf{w}_j = s_2^{d+1}$. (Out of the elements $(\mathbf{y},\mathbf{z})$ of $\mathbf{w}_j$ the transition outputs the elements $(y_0,\dots,y_{m-2d-1},z_0,\dots,z_{d-1})$ and the other elements of $(\mathbf{y},\mathbf{z})$ overlap with the elements of the predecessor state.) Let $t, 1\le t\le nK$, denote the index of $(\mathbf{y},\mathbf{z})_t$. Let $(\cdot,\mathbf{z})_t$ denote the $\mathbf{z}$ part of the symbol $(\mathbf{y},\mathbf{z})_t$ with the following exceptions: We define $(\cdot,\mathbf{z})_{iK} = \mathbf{x}_{i+1}$ for $i=0,\dots,n-1$ corresponding to the first column which is the left border of the interior. Further we note that $\mathbf{w}_{t-K}$ is the predecessor state of $\mathbf{w}_t$.

Let $P((\mathbf{y},\mathbf{z})_t|(\cdot,\mathbf{z})_{t-1},\mathbf{w}_{t-K})$ denote the probability of $(\mathbf{y},\mathbf{z})_t$ conditioned on the causal part of the transition of the process $\mathbf{W}_i$ it is part of. This may be seen as conditioning the new elements of $(\mathbf{y},\mathbf{z})_t$ on the causal elements of the bit-stuffing. By drawing the symbol $(\mathbf{y},\mathbf{z})_{t+1}$ conditioned on $((\cdot,\mathbf{z})_t,\mathbf{w}_{t+1-K})$ using the conditional probability $P((\mathbf{y},\mathbf{z})_t|(\cdot,\mathbf{z})_{t-1},\mathbf{w}_{t-K})$ one can then fill out the interior $B_{n,K)}^*$ one row at a time given the boundaries $B_n^{(v)}$ and $B_K^{(h)}$.

Hence a measure $\mu_{B_{n,K}}$ on $B_{n,K}$ is defined by

$$\begin{aligned}\mu_{B_{n,K}}(\mathbf{w}_0^{n,K-1}) = \pi_X(\mathbf{x}_0)&\\ \cdot\,\mu^v(\mathbf{x}_1,\dots\mathbf{x}_n|\mathbf{x}_0)&\\ \cdot\,\mu^h(\mathbf{y}_0,\mathbf{z}_0,\dots\mathbf{y}_{K-1},\mathbf{z}_{K-1}|\mathbf{x}_0)&\\ \cdot\prod_{t=1}^{nK}P((\mathbf{y},\mathbf{z})_t|(\cdot,\mathbf{z})_{t-1},\mathbf{w}_{t-K})&\end{aligned}\tag{14}$$

where t is the index traversing the interior, row by row.

The entropy of the modified bit-stuffing scheme for the 2-D RLL(d,∞) constraint may be expressed as follows.

Theorem 3.5: The modified bit-stuffing scheme (11), (12) of a given 2-D RLL(d,∞) constraint is parameterized by the width, $m\ge 3d$, of W and the conditional probabilities, $p_1(l), 0\le l<m-d$. The transition probability matrices (10), $\mathbf{P}_m$ and $\mathbf{P}_d$, of the processes $\mathbf{W}$ and $\mathbf{X}$ are determined by bit-stuffing (12) a row of W and X, respectively. The states of $\mathbf{W}$ and $\mathbf{X}$ are $d\times m$ and $d\times d$ elements, respectively. Consider the measure $\mu_{B_{n,K}}$ (14), on the rectangle $B_{n,K}$, induced by the modified bit-stuffing scheme based on $\mathbf{P}_m$ and $\mathbf{P}_d$ and the stationary distributions, π_W and π_X. Based on $\mu_{B_{n,K}}$, the per symbol entropy of the interior $B_{n,K}^*$ of $B_{n,K}$, given the boundary, $B_n^{(v)}\cup B_K^{(h)}$ is given by

$$C_{mb}(d,\infty) = \frac{H_W(m)-H_X(d)}{m-d}, m\ge 3d, \tag{15}$$

where $H_W(m)$ and $H_X(d)$ are the entropies per row (8) determined by $\mathbf{P}_m$ and $\mathbf{P}_d$, respectively, defined by the bit-stuffing.

Proof: Consider the Markov process $\mathbf{W}$ defined by the modified bit-stuffing of W. By Lemmas 3.1 and 3.4, $\mathbf{W}$ is well defined and the stationary solution π_W exists. Due to the reordering and setting $m\ge 3d$, the border processes $\mathbf{X}$ and $\mathbf{Z}$ of the modified bit-stuffing are independent. By Lemmas 3.1 and 3.4, the processes $\mathbf{X}$ and $\mathbf{Z}$ are also well defined and the stationary solutions π_X and π_Z exist. Using the same order (left-to-right) and the same conditional probabilities for $\mathbf{X}$ and $\mathbf{Z}$ when defining p_{ij} (13) ensures that the stationary probabilities are the same, i.e., $\pi_X = \pi_Z$. Initializing the boundary according to these identical stationary distributions, ensures $\mu_X = \mu_Z$ (5).

Given $\mathbf{W}$, the measure $\mu_{B_{n,K}}$ is defined on W_0^{K-1}, where the border Z_{i-1} serves as the border X_i of W_i. Given a configuration on the border X_i, the bit-stuffing on Y_{i-1} and Y_i is independent as the width of X_i is $d = M-1$, and the bit-stuffing of elements of X_i in a given row is performed prior to bit-stuffing of elements of Y_{i-1} and Y_i of that row.

Each element of X_i and Z_i is written prior to any element in Y_i which coincides with the 0-neighborhood of the elements of X_i and Z_i. Therefore any given $\mathbf{W}_i$ is independent of all $\mathbf{X}_j, j\notin\{i,i+1\}$ and the bit-stuffing scheme defines a finite-state Markov description of $\mathbf{W}_i$.

The same conditional probabilities given by, $p_1(l) = p_1(l\mathrm{modulo}(m-d))$ are applied for all $\mathbf{W}_i$. Thus the transfer matrix $\mathbf{P}_m$ is identical for all $\mathbf{W}_i$. The measure $\mu_{B_{n,K}}$ (14) is based on $\mathbf{W}$, including the initialization by the stationary solution, thus all $\mathbf{X}_i$ have identical measures, $\mu_{X_i} = \mu_X, 0\le i\le K-1$. Initializing the boundary based on the stationary solution π_W for all $\mathbf{W}_i$, gives $\mu_{W_i} = \mu_W, 0\le i\le K-1$.

Consider the elements $(\mathbf{y},\mathbf{z})_t$, on Y_i as one symbol as in (14). Because of the independence of the $\mathbf{X}_i$s the conditional probability of an element $(\mathbf{y},\mathbf{z})_t$ is given by the process $\mathbf{W}_i$ it belongs to. The conditional probability is given by one transition of $\mathbf{W}_i$ where all prior elements coinciding with a 0-neighbor element of an element in $(\mathbf{y},\mathbf{z})_t$ are part of the predecessor state or one of the new elements in $\mathbf{X}_i$ or $\mathbf{Z}_i$ of the transition.

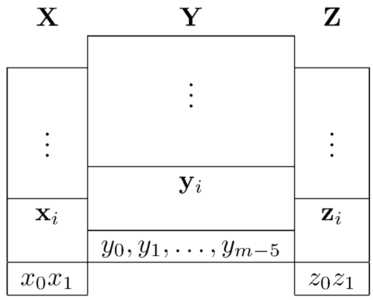

Fig. 4. The modified bit-stuffing scheme for the $\diamond(3)$ constraint. The state $\mathbf{w}_i$ consists of the three blocks $\mathbf{x}_i, \mathbf{y}_i$ and $\mathbf{z}_i$, each having a height of two rows. On the transition to state $\mathbf{w}_j$ the new elements $x_0, x_1, z_0, z_1, y_0, \ldots, y_{m-5}$ are output.

For notational consistency, $\mathbf{w}_t$ is written as $(\cdot, \mathbf{z})_{t-1}, (\mathbf{y}, \mathbf{z})_t$ when representing it in a decomposed form. The conditional probability derived from one transition within some $\mathbf{W}_i$ is

$$\begin{aligned} P(\mathbf{w}_t|\mathbf{w}_{t-K}) &= P((\cdot,\mathbf{z})_{t-1}|\mathbf{w}_{t-K}) \cdot P((\mathbf{y},\mathbf{z})_t|((\cdot,\mathbf{z})_{t-1}, \mathbf{w}_{t-K}) \\ &= P((\cdot,\mathbf{z})_{t-1}|(\cdot,\mathbf{z})_{t-1-K}) \cdot P((\mathbf{y},\mathbf{z})_t|(\cdot,\mathbf{z})_{t-1}, \mathbf{w}_{t-K}) \end{aligned} \tag{16}$$

as $(\cdot, \mathbf{z})_{t-1}$ only depends on $(\cdot, \mathbf{z})_{t-1-K}$ of $\mathbf{w}_{t-K}$. The terms are given by a stationary solution, $\mu_{W_i} = \mu_W$, (on W_i) independent of the value of the index t within the interior.

Therefore solving for the last term of (16) and calculating the conditional entropy of the new elements, $(\mathbf{y}, \mathbf{z})_t$, of one transition gives

$$H_W(m) - H_X(d). \tag{17}$$

Dividing by the number of elements, $(m-d)$, of $(\mathbf{y}, \mathbf{z})_t$, gives (15). ■

Considering the combined symbols $(\mathbf{y}, \mathbf{z})$ the measure (14) on the interior provides a stationary description. Viewing the individual elements in $(\mathbf{y}, \mathbf{z}_t)$ the measure may be seen as quasi-stationary.

B. Bit-Stuffing for Checkerboard Constraints

In the previous subsection, bit-stuffing was modified and analyzed for 2-D RLL(d, ∞) constraints. Since bit-stuffing including the modified bit-stuffing scheme is applicable for all checkerboard constraints, a more general presentation of the modified bit-stuffing is given, taking the point of departure in the diamond constraint $\diamond(M)$. The main difference is that for a transition, the elements of Y_i may be offset a few rows back compared to the elements of X_i and Z_i.

Example 3.6: Consider modifying bit-stuffing for the $\diamond(M)$ constraint. The width of the borders is chosen as $b = M - 1$. To maintain independence of the borders of the process $\mathbf{W}$, the elements of a new row of X and Z must be written prior to a new row of Y which is at least $M - 1$ rows back (Fig. 4). This is obtained by writing the new elements of X and Z before the new elements of Y in one transition and letting these new elements of Y be $M - 2$ rows after the row with new elements of X and Z. This is to ensure that the 1-norm distance between the last element, x_{b-1}, of the new row of X and the first element of the old row of Y is M such that no element of Y in the old row will influence the new elements in X (Fig. 4). Besides this modification, the bit-stuffing scheme and the calculation of $\mathbf{P}_m$ may proceed as for the 2-D RLL(d, ∞) constraint. The transition probabilities, (p_{ij}), are defined by a product of conditional probabilities (10) derived from the bit-stuffing probabilities, $p_1(l)$, using (12).

In the general case let the extent of the checkerboard constraint be $N \times M$. We shall introduce the modified bit-stuffing and the corresponding Markov process jointly. Let b be the width of X and Z. Let $x_0, \ldots, x_{b-1}, z_0, \ldots, z_{b-1}, y_0, \ldots, y_{m-2b-1}$ be the order of the new elements of state $\mathbf{w}_j$ in the transition from the predecessor state $\mathbf{w}_i$. The new elements of X and Z belong to the same row, whereas the new elements of Y may be positioned a few rows behind. Let S_b specify the number of rows the new elements of Y are behind. Let S_t denote the number of rows the elements of a state of $\mathbf{W}$ are defined on. Three conditions on the set of elements defining the states of the Markov process $\mathbf{W}$ are given:

C.1. $m \geq 2b + M - 1 \wedge b \geq M - 1$.

C.2. Positioning the 1 of the 0-neighborhood $\mathcal{N}$ at each position of $(x_0, \ldots, x_{b-1})$ and $(z_0, \ldots, z_{b-1})$, no 0 of $\mathcal{N}$ may coincide with a causal element of Y, but all causal elements of X and Z, respectively, are part of the state.

C.3. Positioning the 1 of the 0-neighborhood $\mathcal{N}$ at each position of $(y_0, \ldots, y_{m-2b-1})$, the state transition includes all elements coinciding with a 0 of $\mathcal{N}$, which are either an element of X or Z or a causal element of Y.

C.4. Consider one transition, in each column the new element and the elements of the old state of the column must form a contiguous set.

Thus the conditions above specify a minimum set of elements, which must be included in the state of $\mathbf{W}$ for a given checkerboard constraint and requirements for the elements of one transition. Theorem 3.5 is formulated for the minimum set of elements for 2-D RLL(d, ∞).

For any checkerboard constraint of extent (N, M), the state specified below will satisfy the conditions. Let the width of the borders X and Z be $b = M - 1$ and the width, m, of W be a value which satisfies C.1. In order to avoid any influence of elements of Y (C.2) on the writing of elements on X (and Z), $(x_0, \ldots, x_{b-1})$ of $\mathbf{X}$ (and $(z_0, \ldots, z_{b-1})$ of $\mathbf{Z}$) are written $N - 1$ rows ahead of writing $(y_0, \ldots, y_{m-2b-1})$ on Y. The state of $\mathbf{W}$ is given by the union of the following elements: 1) The $N - 1$ elements above each of the elements $(y_0, \ldots, y_{m-2b-1})$ due to the last part of C.3. 2) The $2N - 1$ elements above each of the elements $(x_0, \ldots, x_{b-1})$ and $(z_0, \ldots, z_{b-1})$. These are included to ensure C.2 and the first part of C.3.

Based on $\mathbf{W}$ satisfying C.1-4, the bit-stuffing is extended to the set of elements, $D_{n,K}$ defined by the extension from W to W_0^{K-1} (6). Assume the initial state of each $\mathbf{W}_i$ is given. The configuration on the segment $D_{n,K}$ is determined by the set of initial states and n transitions of each $\mathbf{W}_i$ on W_0^{K-1}. The segment $D_{n,K}$ has width $K(m-b)+b$. The difference compared to the array $B_{n,K}$ is that the height may vary according to whether the column belongs to X_i or Y_i. (Columns of Z_i have the same height as columns of X_i.) The height in column l is given by n plus the height of the state in column l. Given the initial states the bit-stuffing scheme addresses the elements of $D_{n,K}$ in the order given by one transition of each $\mathbf{W}_i$ in order of increasing i before proceeding to the next sequence of transitions. Within each transition the elements of Z_i is addressed prior to the elements of Y_i.

This modified bit-stuffing scheme defines the transition probabilities p_{ij} (10) based on the conditional probabilities, $p(R_l = r_l | C(l) = c(l))$. When a 1 is admissible given context $c(l)$ the conditional probabilities must satisfy $0 < p(R_l = r_l | C(l) = c(l)) < 1$.

The four conditions, C.1-4, above define restrictions on the states of $\mathbf{W}$. These are supplemented by two conditions on the contexts of the conditional probabilities.

C.5. The contexts $c(l)$ must include the causal 0-neighborhood elements of the given $\mathbf{W}_i$ at position l.

When the context elements of $c(l)$ are exactly given by the causal 0-neighborhood elements and the conditional probabilities by $p_1(l)$ (12), the bit-stuffing is called a *a modified bit-stuffing scheme for a checkerboard constraint*. The modified bit-stuffing for 2-D RLL(d, ∞) is an example of this.

The contexts may also be extended and include more elements than just those defining whether $r_l = 1$ is admissible. In this case, we call it a context-based bit-stuffing scheme.

C.6. For $\mathbf{X}$ (and $\mathbf{Z}$) the contexts are mappings of causal elements of $\mathbf{X}$ (and $\mathbf{Z}$) within the states of the corresponding transition. For the elements of Y the contexts are mappings of the causal elements of the transition, i.e., elements of the old state and the new elements of $\mathbf{X}$ and $\mathbf{Z}$ and prior new elements of Y. The context mappings for column l in $\mathbf{X}$ and column $m-b+l$ in $\mathbf{Z}, 0 \le l < b$ are identical and the same set of conditional probabilities are applied. (This is to ensure the independence (5)).

Definition 3.7: For a given checkerboard constraint, consider a Markov process $\mathbf{W}$, of width m, having states satisfying C.1-4 and transition probabilities derived from a set of conditional probabilities satisfying C.5-6. Assume the values of the elements of the initial states of all $\mathbf{W}_i$ on W_0^{K-1} are given. The *context based bit-stuffing scheme* on $D_{n,K}$ based on $\mathbf{W}$ is defined by a set of conditional probabilities $p(R_l = 1|C(l') = c(l)), l' = l\text{modulo}(m-b)$ satisfying C.5-6. The order in which the elements of $D_{n,K}$ are addressed is given by a transition of each $\mathbf{W}_i$ in order of increasing i. Within each transition the order is given by the reordering (9).

The independence of the borders (7) is ensured by the offset of the new row of $\mathbf{Y}$ and $m \ge 2b + M - 1$. The conditions C.1-2 ensures that $\mathbf{Y}_{i-1}$ and $\mathbf{Y}_i$ are independent given the output of $\mathbf{X}_i$.

The conditional probabilities of the context-based bit-stuffing scheme for checkerboard constraints defines the probability matrix $\mathbf{P}_m$ of $\mathbf{W}$. The stationary distribution is given by $\pi_W \mathbf{P}_m = \pi_W$. The construction of $\mu_{B_{n,K}}$ (14) for the 2-D RLL(d,∞) constraint is generalized to a measure $\mu_{D_{n,K}}$ on $D_{n,K}$, which in the same manner is based on $\mathbf{W}$ and the transition matrix $\mathbf{P}_m$. The stationary distribution π_W is used to initialize the initial states of all $\mathbf{W}_i$ on W_0^{K-1} defining a measure on the upper horizontal boundary. The measure on the left vertical boundary is defined by $\mathbf{X}_0$ and the stationary distribution π_X. The probabilities on the interior, $D_{n,K}^*$, is given by the conditional probabilities of $(\mathbf{y}, \mathbf{z}_t)$ as for $B_{n,K}$.

The context based bit-stuffing thus defines a sequence of (2-D) probability measures $\mu_{D_{n,K}}$ indexed by n and K. By construction, the measure $\mu_{D_{n,K}}$ is obtained as the marginal measure on $D_{n,K}$ of any $\mu_{D_{n'',K''}}$, where $n \le n''$ and $K \le K''$. Thus the measure is nested. Let $H(\mu_{D_{n,K}})$ denote the entropy of the measure $\mu_{D_{n,K}}$. The nesting property allows us to take the limit [4].

Let n' and m' denote the size of the sides of the bounding box of the elements of $D_{n,K}$. (The elements of the bounding box, which are not in $D_{n,K}$ are set to 0.)

Definition 3.8: The capacity, $C_C(m,b)$ of the context based bit-stuffing for a checkerboard constraint is defined as

$$C_C(m,b) = \lim_{n,K\to\infty} \frac{H(\mu_{D_{n,K}})}{n'm'} \tag{18}$$

where the sides of the bounding box $n', m' \to \infty$ as $n, K \to \infty$.

Theorem 3.9: Consider a given checkerboard constraint of extent (N, M). Assume that the finite state Markov process $\mathbf{W}$ has states and conditional probabilities satisfying the conditions, C.1-6 with $b \ge M-1$ and $m \ge 2b + M - 1$. A context based bit-stuffing scheme for the checkerboard constraint based on $\mathbf{W}$ has the capacity

$$C_C(m,b) = \frac{H_W(m) - H_X(b)}{m-b}, \tag{19}$$

where $H_W(m)$ and $H_X(b)$ are the entropies per row (8) of the processes $\mathbf{W}$ and $\mathbf{X}$.

Proof: The first step of the proof follows the proof of Theorem 3.5 for a given n and K. $\mathbf{W}$, $\mathbf{X}$ and $\mathbf{Z}$ are finite state Markov processes. Again by Lemmas 3.1 and 3.4, $\mathbf{W}$ and the borders $\mathbf{X}$ and $\mathbf{Z}$ are all well defined and have stationary solutions. The conditions C.1-3 maintain the independence of border processes, $\mathbf{X}$ and $\mathbf{Z}$. Using the same context mappings and conditional probabilities in the bit-stuffing scheme for all $\mathbf{W}_i$ gives $\mu_{W_i} = \mu_W$.

Consider the output of transition j of each process $\mathbf{W}_i, 0 \le i < K$, which overlap such that the configuration on X_i is given by the output of $\mathbf{Z}_{i-1}$. The elements output by this set of transitions, correspond to one row of elements on $B_{n,K}$ in Theorem 3.5, but now possibly with the new elements of X_i and Z_i in one row and the the new elements of Y_i in another row. The interior $D_{n,K}^*$ of $D_{n,K}$ is given by the output of these transitions for $1 \le j \le n$, where the elements of the first b columns, i.e., the left vertical boundary, are given by $\mathbf{X}_0$. The distribution on the boundary, i.e., the initial states of $\mathbf{W}_i, 0 \le i < K$ and the b first columns, is initialized based on the stationary distribution, π_W. Thus the boundary is intialized based on the stationary distributions in the same manner as for $\mu_{B_{n,K}}$. As in Theorem 3.5, the stochastic variables $\mathbf{Y}_{i-1}$ and $\mathbf{Y}_i$ are independent given the output of $\mathbf{X}_i$, as $b \ge M-1$ and the contexts of $\mathbf{X}_i$ do not have any elements of Y_{i-1} and Y_i. Likewise $\mathbf{W}_i$ is independent of all $\mathbf{X}_j, j \notin \{i, i+1\}$. By definition of the bit-stuffing scheme, $\mathbf{P}_m$ is identical for all $\mathbf{W}_i$. This leads to identical stationary distributions on the initial states, which in turn gives $\mu_{W_i} = \mu_W, 0 \le i < K$.

The arguments of Theorem 3.5 still hold under these generalizations for each transition of p_{ij}. The probability of the elements $(\mathbf{y}, \mathbf{z})_t$ conditioned on the causal elements is given by the identical and stationary processes $\mathbf{W}_i$. Therefore the contribution to the entropy for each $(\mathbf{y}, \mathbf{z})_t$ conditioned on the causal elements is as in (16) given by

$$H_W(m) - H_X(b), \quad m \ge 2b + M - 1. \tag{20}$$

The second step of the proof shall show that asymptotically, the expression (20) determines the capacity. Let n^* and $m^* = K(m-b)$ denote the sides of the largest rectangle defined on the interior, $D_{n,K}^*$. The entropy $H(\mu_{D_{n,K}})$ relative to the size of the bounding box with sides n' and m' is bounded by

$$\begin{aligned} \frac{H_W(m) - H_X(b)}{m-b}\frac{n^*m^*}{n'm'} &\le \frac{H(\mu_{D_{n,K}})}{m-b}\frac{n^*m^*}{n'm'} \le \\ \frac{H_W(m) - H_X(b)}{m-b}\frac{n^*m^*}{n'm'} &+ \frac{n'b + m'(S_t + S_b)}{n'm'}|A| \end{aligned} \tag{21}$$

where S_t is the height of the states, S_b is the number of rows which X extends below Y, and $|A|$ is the size of the alphabet.

As $m^* = m' - b$ and $n^* = n' - S_t - S_b$ and since m, b, S_t, S_b and $|A|$ are all fixed values, asymptotically both the lower and upper bound in (21) converge to

$$\frac{H_W(m) - H_X(b)}{m-b}, \quad \text{as } n, K \to \infty. \qquad \blacksquare$$

Theorem 3.9 also applies to the modified bit-stuffing scheme as this is a special case of the context based bit-stuffing for checkerboards constraints. Actually the assumption of initialization based on the stationary distribution, π_W is not necessary as each $\mathbf{W}_i$ will converge to the stationary solution for $n \to \infty$.

C. Optimizing the Entropy

The modified bit-stuffing scheme introduced is completely predefined in the sense that the use of biased sequences is decided *a priori* based on the column number and given by $p_1(l)$. For context based bit-stuffing, an increased number of biased sequences may be used. The decision of which biased sequence and thereby the conditional probability to be used may depend on a larger context.

Example 3.10: For the $\diamond(3)$ constraint, we applied context based bit-stuffing choosing the bit-stuffing probabilities p_1 conditioned on the values of the elements in the previous state. The states of $\mathbf{W}$ were defined as depicted on Fig. 4. The next row of the processes $\mathbf{X}$ (and $\mathbf{Z}$) was specified according to probabilities conditioned on the two previous rows. These processes were chosen such that they were symmetric in the two columns. The elements of the new row of Y was then specified according to probabilities conditioned on all three rows of the transition of $\mathbf{X}$ and $\mathbf{Z}$ combined with the two rows of the predecessor state on Y. These conditional probabilities for the new elements $(y_0, \ldots, y_{m-2b-1})$ on Y were obtained from the transition probabilities of the maxentropic solution [10] for $\mathbf{W}$ derived from the transfer matrix, $\mathbf{T}_m$. Thus, p_{ij} was specified directly based on a product of the conditional probabilities of the three sets $(x_0, \ldots, x_{b-1}), (z_0, \ldots, z_{b-1})$, and $(y_0, \ldots, y_{m-2b-1})$ generated in one transition.

D. Some Practical Remarks

A drawback of bit-stuffing is the fact that it is a variable length code and it has error propagation. The use of independent borders, in modified bit-stuffing, introduces a block structure in the horizontal direction which may be useful in handling error propagation.

Writing data of the modified bit-stuffing scheme row by row introduces a latency of $m-d-1$ elements if $z_0, \ldots z_{d-1}$ is written before y_i. This latency may be reduced to d if the writing of z_l and y_l elements are interleaved.

For 2-D RLL(d, ∞), the plane may be written column by column instead of row by row, using the biased sequence designated to the column. Thus the choice of biased sequence is only changed once for each new column. The column with z_i must be written before the column with y_{m-3d+i} in this case.

IV. Deterministic Borders

As a special case consider fixed configurations on the borders, i.e., $\mathbf{X}$ and $\mathbf{Z}$ are deterministic. This could be motivated by a desire of having a synchronization component or by classes of constraints for which bit-stuffing is not readily applicable. An example of the latter is the symmetrical RLL, 2-D SRLL(d, k), constraint [14], where the run-lengths limits apply to all symbols in the alphabet. For this constraint, methods for constructing configurations on a merging array, Y, given any two admissible configurations, on the arrays X and Z, is given in [14]. The (minimum) width of the merging array, Y, is given as a function of the d parameter.

The existence of a solution in between two given configurations on the arrays is a necessary but not sufficient prerequisite for applying a modified bit-stuffing scheme. It is not clear to what extent modified bit-stuffing is applicable to a SRLL(d, k) constraint.

However, using a deterministic and periodic configuration, $\mathbf{b}$, to define the borders, $\mathbf{X} = \mathbf{b}$ and $\mathbf{Z} = \mathbf{b}$, one can consider $\mathbf{bYb}$ a stochastic variable and determine the max-entropic solution for $\mathbf{Y}$ conditioned on $\mathbf{X} = \mathbf{b}, \mathbf{Z} = \mathbf{b}$ and in this way obtain a lower bound on the entropy of the constraint using (4). In order to facilitate the use of a finite state source description periodic deterministic borders shall be used.

A. Periodic Borders

Looking beyond checkerboard constraints, consider an arbitrary constraint of extent $N \times M$. Consider a 2-D configuration $\mathbf{b}$ on an array of width w and any fixed height. This array is repeated at intervals of $m + w$ columns horizontally. This leaves arrays of width m undefined in between the repetition of two borders.

Definition 4.1: Let the configuration $\mathbf{b}$ of width w be periodic vertically with the period p. The periodic array $\mathbf{b}$ is a *periodic merging array* if for any arrays $\mathbf{X} = \mathbf{x}$ and $\mathbf{Y} = \mathbf{y}$ for which $\mathbf{bxb}$ and $\mathbf{byb}$ are admissible configurations according to the constraint, the cascaded array $\mathbf{bxbyb}$ is also admissible.

Given a periodic merging array, $\mathbf{b}$, the array $\mathbf{bYb}$ is called a $\mathbf{b}$-boundary constrained array.

For a given $\mathbf{b}$ with $w \geq M-1$, any pair of admissible configurations, $\mathbf{bxb}$ and $\mathbf{byb}$, may be cascaded to form $\mathbf{bxbyb}$. Thus these arrays are $\mathbf{b}$-boundary constrained arrays. The reason is that the merging array separates the arrays X and Y in the sense that no $N \times M$ rectangle contains symbols from both X and Y. Further they are bounded by identical merging arrays. For some constraints, a periodic merging array, $\mathbf{b}$, of width $w < M-1$ may be specified such that the property of the $\mathbf{b}$-boundary constrained array still holds.

The admissible configurations of $\mathbf{bYb}$ may be described by a finite state source with states of height $n \geq N-1$ and width $m + 2w$. (Viewing the states on a cylinder, the width may be reduced to $m + w$). For a merging array period $p \leq n+1$, the phase of the period is contained in one transition in the elements of the states given by symbols of $\mathbf{b}$. For $p \geq n+1$ the phase information could still be uniquely defined by the states. If this is not the case the (missing) information about the phase has to be added to the states.

The boundary $\mathbf{b}$ is periodic and hence deterministic, yielding an entropy which is zero. Therefore, using (4) to calculate the entropy gives the following lower bound on the entropy of the constraint:

$$H^{(2)} \geq \frac{H_{bYb}(m)}{m+w} \tag{22}$$

where H_{bYb} is the per row combinatorial entropy (3) of $\mathbf{bYb}$.

This approach was explored in [15] where entropy results for the SRLL(2,3) constraint were presented. These results were comparable to what may be achieved based on Etzion's merging technique [14] combined with (4). Below the technique of periodic borders shall be applied to another constraint.

B. Density Constraints

Another class of constraints is given by imposing constraints on the local average value or density of given values of the alphabet. This constraint has some resemblance to halftoning, i.e., rendering gray scale images by binary images. Let x_{ij} be the symbols within any $N \times M$ rectangle.

Definition 4.2: Given N and M, the binary density constraint with parameters $d_{\min}$ and $d_{\max}$ is defined by

$$d_{\min} \leq \sum_{i=1}^{N} \sum_{j=1}^{M} x_{ij} \leq d_{\max} \tag{23}$$

where $0 \leq d_{\min} \leq d_{\max} \leq NM$ and $x_{ij} \in \{0, 1\}$.

Let $r\mathbf{x}$ denote an $N \times M$ configuration given by row r followed by $\mathbf{x}$ having $N-1$ rows. Likewise let $\mathbf{x}r$ denote $\mathbf{x}$ followed by r. For the density constraint, given any admissible configuration $r\mathbf{x}$, the configuration $\mathbf{x}r$ is also admissible, as the sums are identical. By repeating the argument in both directions it is clear that any admissible configuration on an $n = N$ by $m = M$ rectangle defines the period of a doubly periodic admissible configuration. Therefore any admissible configuration defined on an $N \times (M-1)$ rectangle may be used as the period of a merging array.

Given configurations on the arrays X and Z it is not clear what the width of a merging array between these should be for the density constraint.

Example 4.3: For the $(d_{\min}, d_{\max}) = (4,5)$ and $N = M = 3$ density constraint, it may be shown that there are no solutions for $w \leq 7$. Consider the configuration

$$\begin{array}{cccccccc} \multicolumn{2}{c}{\mathbf{X}} & & & \mathbf{Y} & & \multicolumn{2}{c}{\mathbf{Z}} \\ 0 & 0 & y_{11} & y_{12} & \dots & y_{17} & 1 & 1 \\ 0 & 0 & y_{21} & y_{22} & \dots & y_{27} & 1 & 1 \\ 1 & 0 & y_{31} & y_{32} & \dots & y_{37} & 1 & 0 \end{array}$$

Due to the constraint and the configuration $\mathbf{x}$, the first column, (y_{11}, y_{21}, y_{31}), of Y must be all 1. Hence $y_{12}, y_{22}, y_{32}, y_{13}, y_{23}, y_{33}$ will contain at least four 0s and at most two 1s. Thus the fourth column has to contain at least two 1 elements, (y_{i4}). On the other hand, the seventh column of Y must be all 0. Hence columns five and six, (y_{15}, y_{25}, y_{35}) and (y_{16}, y_{26}, y_{36}), will contain at least four 1s and at most two 0s. Thus the fourth column has to contain at least two 0 elements, leading to a contradiction. This example gives an instance for which no merging array of width seven exists. This and other similar examples show that no merging array of width seven or less exists for this constraint.

C. Cyclic Matrices

As mentioned the use of periodic merging arrays introduces a period p and phases in the finite state source. The states may be grouped in classes according to their phase. With a proper ordering this means that each transition is from a state in class i to one in class $i+1$ modulo p. With a proper ordering of indices the transfer matrix $\mathbf{T}_m$ is block cyclic. Let $\lambda_m = |\Lambda_m|$, where Λ_m is an eigenvalue of $\mathbf{T}_m$ with maximum magnitude. The matrix after p transitions, $\mathbf{T}_m^p$, has a block diagonal form with p blocks namely a block for each phase. By Perron–Frobenius [16], λ_m^p is p-tuple eigenvalue for $\mathbf{T}_m^p$ and $\mathbf{T}_m$ has p simple eigenvalues $\lambda_m e^{ik2\pi/p}$, $k = 0, \dots, p-1$. (In special cases the multiplicity is a multiple of p.) This structure may be taken into consideration in the calculation of λ_m determining the capacity.

Finally it may be remarked that there are constraints for which it is clear that one cannot find a finite merging array for any two valid arrays. The standard example is domino tiling, where the whole plane is tiled by one by two vertical and horizontal domino pieces. Consider the case where a configuration $\mathbf{x}$ has a vertical zig-zag boundary of all horizontal pieces where the piece in every other row is displaced one position relative to its two neighbors. In this case there is only one solution extending off the boundary of $\mathbf{x}$, namely that which locks up with the boundary, except for the first and last piece of the zig-zag. (By induction it is seen that for any finite width merging array, a long enough zig-zag boundary can lead to a conflict with the opposing boundary.)

V. Numerical Results

Results are presented for the modified bit-stuffing scheme. The following examples are considered: Three instances of the 2-D RLL(d,∞) constraint, $d = 2, 3$, and 4, as well as the $\diamond(3)$ constraint. Thereafter the lower bound on the entropy of the density constraint for $(d_{\min}, d_{\max}) = (4,5)$ and $N = M = 3$ obtained using periodic merging arrays is presented.

A. Modified Bit-Stuffing Scheme

Table I presents entropy results (15) for applying the modified bit-stuffing scheme (11), (12) to the checkerboard constraints in consideration. The width of the band used is also given. For a given width m, the modified bit-stuffing scheme is specified by the parameters, $p_1(l)$. The values of the parameters, $p_1(l)$, used to obtain the results in Table I are listed in the Appendix. Starting with one unbiased sequence and increasing the number of parameters used, the following entropies (and thereby lower bounds) were calculated. $H_{p=1/2}$ gives the entropy using an unbiased bit-stuffer. H_p is the optimized entropy over a single biased sequence, whereas H_{p_X,p_Y} is optimized choosing two different biased sequences: One for the borders, X and Z, and one for the interior, Y. A slight improvement of the lower bound $H_{\mathbf{p}\mathrm{opt}}$ was found by using a different biased sequence for each column $l, 0 \leq l < m-d$, optimizing the entropy. The optimization was performed using a steepest descent approach, viewing the entropy as a function of the conditional probabilities (12) indexed by the column, $(p_1(0), \dots, p_1(m-d-1))$, and searching in the direction of the gradient.

TABLE I
ENTROPY OF THE BIT-STUFFING INDUCED MEASURES USING DIFFERENT BIASED SEQUENCES. THE FINAL COLUMN H_U PROVIDES AN UPPER BOUND ON THE ENTROPY OF THE CONSTRAINTS

	m	$H_{p=1/2}$	H_p	H_{p_X,p_Y}	$H_{\mathbf{p}opt}$	H_U
RLL$(2,\infty)$	19	0.3917	0.4398	0.4410	0.4415	0.4459
RLL$(3,\infty)$	16	0.3050	0.3606	0.3628	0.3641	0.3686
RLL$(4,\infty)$	15	0.2487	0.3082	0.3110	0.3123	0.3188
$\diamond(3)$	14	0.2764	0.3441	0.3466	0.3478	0.3542

Finally H_U gives an upper bound on the entropy using the methods of [17]. These offer an improvement over the simple upper bound given by (3).

It should be noted that the upper and lower bounds are quite close to each other, that is their relative difference is in the range of 1–2%.

For comparison the lower bounds on the entropy of conventional bit-stuffing for 2-D RLL(d,∞) presented by Halevy *et al.* [11] are $0.4267, 0.3402$, and 0.2858 for $d = 2, 3$, and 4, respectively. Simulation results [18], based on performing bit-stuffing, supports a conjecture that the entropy of conventional bit-stuffing is slightly greater than that of the modified bit-stuffing. If this is the case, the $H_{\mathbf{p}\mathrm{opt}}$ provides a new lower bound on the entropy of conventional bit-stuffing.

Following the description in Section III-C, Example 3.10, the context based bit-stuffing method was used to improve the lower bound to 0.3497 for the $\diamond(3)$ constraint.

B. Density Constraint $(4,5)$

Table II gives the capacity using a specific periodic merging array for the density constraint for $(d_{\min}, d_{\max}) = (4,5)$ and $N = M = 3$ for some values of the width of Y, m. The chosen merging array has period $p = 2$, and the period is given by 10 and 01 in every other row. The results provide a lower bound on the capacity of the constraint. An upper bound (3) on the capacity of 0.554 is obtained using an array of width $m = 11$.

TABLE II
LOWER BOUNDS ON CAPACITY FOR THE DENSITY(4,5) CONSTRAINT

m	$\frac{H(m)}{m+8}$	$\frac{H_{bYb}(m)}{m+2}$
7	0.269	0.353
8	0.280	0.360
9	0.289	0.372
10	0.297	0.382

The minimum width for which some merging array exists for any arbitrary pair of configurations on the arrays X and Y is not known to us. Example 4.3 showed that it is at least 8. The results inserting this value for c in the lower bound (4) on the entropy is also given in Table II for comparison. It is seen that the periodic merging array approach provides significantly higher entropy than what can be hoped

TABLE III
THE BIT-STUFFING PROBABILITIES USED IN TABLE I TO OBTAIN H_p AND H_{p_X,p_Y}, RESPECTIVELY. THE WIDTH m IS ALSO GIVEN FOR EACH OF THE CONSTRAINTS RLL$(2,\infty)$, RLL$(3,\infty)$, RLL$(4,\infty)$ AND $\diamond(3)$

	m	p	p_X	p_Y
RLL$(2,\infty)$	19	0.289	0.220	0.297
RLL$(3,\infty)$	16	0.250	0.193	0.271
RLL$(4,\infty)$	15	0.220	0.175	0.285
$\diamond(3)$	14	0.225	0.160	0.245

TABLE IV
THE BIT-STUFFING PROBABILITIES $p_1(l)$ USED FOR COLUMN l TO ACHIEVE $H_{\mathbf{p}\mathrm{opt}}$ FOR THE CONSTRAINTS RLL$(2,\infty)$, RLL$(3,\infty)$, RLL$(4,\infty)$ AND $\diamond(3)$ WITH WIDTHS OF $m = 19, 16, 15$ AND 14, RESPECTIVELY

	RLL$(2,\infty)$	RLL$(3,\infty)$	RLL$(4,\infty)$	$\diamond(3)$
l	$p_1(l)$			
1	0.1968	0.1603	0.1261	0.1497
2	0.2283	0.1841	0.1422	0.1733
3	0.2761	0.2116	0.1669	0.2681
4	0.2774	0.2550	0.1805	0.2302
5	0.2779	0.2554	0.2417	0.2309
6	0.2779	0.2558	0.2478	0.2312
7	0.2778	0.2558	0.2636	0.2312
8	0.2778	0.2558	0.2731	0.2312
9	0.2778	0.2550	0.2733	0.2315
10	0.2778	0.2531	0.2848	0.2307
11	0.2778	0.2635	0.4205	0.2558
12	0.2778	0.2925	0.1261	0.3410
13	0.2780	0.3836	0.1422	0.1497
14	0.2781	0.1603	0.1669	0.1733
15	0.2762	0.1841	0.1805	
16	0.3225	0.2116		
17	0.4153			
18	0.1968			
19	0.2283			

for using a merging array as expressed by inserting $c = 8$ in the lower bound (4).

VI. CONCLUSION

A modified bit-stuffing scheme applicable to 2-D checkerboard constraints was presented. The scheme presented is easy to analyze based on well-known 1-D techniques. A probability measure is derived from the modified bit-stuffing and the entropy of the scheme may be calculated. Numerical entropy results are given for examples of checkerboard constraints, namely 2-D RLL$(d,\infty) d = 2,3,4$ and the constraints given by a min 1-norm distance of 3 between 1s on a binary alphabet. These entropies provide lower bounds on the capacity of the constraints. The entropies obtained are within 1–2% of upper bounds on the capacity. The same construction used for analyzing the modified bit-stuffing was also used for analyzing periodic merging arrays which was applied to a density constraint.

APPENDIX

This Appendix lists the parameters used to obtain the entropy results for the modified bit-stuffing scheme presented in Table I, i.e., for the RLL$(2,\infty)$, RLL$(3,\infty)$, RLL$(4,\infty)$ and $\diamond(3)$ constraints. The parameters determine the bit-stuffing probabilities, $p_1(l)$, applied in column l. Table III lists the single biased bit-stuffing probability, p, leading to the entropy H_p and the two bit-stuffing probabilities, p_X and p_Y, leading to H_{p_X,p_Y}.

Table IV lists the bit-stuff probabilities, $p_1(l)$, applied in column l to obtain $H_{\mathbf{p}\mathrm{opt}}$. These values of $p_1(l)$ were determined by a steepest descent search.

REFERENCES

[1] K. A. S. Immink, P. H. Siegel, and J. K. Wolf, “Codes for digital recorders,” *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2260–2299, Oct. 1998.

[2] P. Vettiger, G. Cross, M. Despont, U. Drechsler, U. Durig, B. Gotsmann, W. Haberle, W. M. Lantz, H. Rothuizen, R. Stutz, and G. Binnig, “The “millipede”-nanotechnology entering data storage,” *IEEE Trans. Nanotechnol.*, vol. 1, no. 1, pp. 39–55, 2002.

[3] E. Eleftheriou, T. Antonakopoulos, G. Binnig, G. Cherubini, M. Despont, A. Dholakia, U. Duerig, M. Lantz, H. Pozidisand, H. Rothuizen, and P. Vettiger, “Millipede-a mems-based scanning-probe data-storage system,” *IEEE Trans. Magn.*, vol. 39, no. 2, pp. 938–945, Mar. 2003.

[4] R. Roth, P. H. Siegel, and J. K. Wolf, “Efficient coding schemes for the hard-square model,” *IEEE Trans. Inf. Theory*, vol. 47, no. 3, pp. 1166–76, Mar. 2001.

[5] S. Friedland, “On the entropy of Z^d subshifts of finite type,” *Linear Algebra Appl.*, pp. 199–220, 1997.

[6] N. G. Markley and M. E. Paul, “Matrix subshifts for Z^v symbolic dynamics,” *Proc. London Math. Soc.*, vol. 43, pp. 251–272, 1981.

[7] A. Kato and K. Zeger, “On the capacity of two-dimensional run-length constrained channels,” *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1527–1540, Jul. 1999.

[8] W. Weeks, IV and R. Blahut, “The capacity and coding gain of certain checkerboard codes,” *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 1193–1204, May 1998.

[9] Z. Nagy and K. Zeger, “Asymptotic capacity of two-dimensional channels with checkerboard constraints,” *IEEE Trans. Inf. Theory*, vol. 49, no. 9, pp. 2115–2125, Sep. 2003.

[10] S. Forchhammer and J. Justesen, “Entropy bounds for constrained two-dimensional random fields,” *IEEE Trans. Inf. Theory*, vol. 45, no. 1, pp. 118–127, Jan. 1999.

[11] S. Halevy, J. Chen, R. M. Roth, P. H. Siegel, and J. K. Wolf, “Improved bit-stuffing bounds on two-dimensional constraints,” *IEEE Trans. Inf. Theory*, vol. 50, no. 5, pp. 824–838, May 2004.

[12] Z. Nagy and K. Zeger, “Entropy bounds for the hard-triangle model,” in *Proc. IEEE Int. Symp. Information Theory*, Chicago, IL, 2004, p. 162.

[13] C. E. Shannon, “A mathematical theory of communication,” *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, Jul. 1948.

[14] T. Etzion, “Cascading methods for runlength-limited arrays,” *IEEE Trans. Inf. Theory*, vol. 43, no. 1, pp. 319–324, Jan. 1997.

[15] S. Forchhammer and T. V. Laursen, “Cascading 2d constrained arrays using periodic merging arrays,” in *Proc. IEEE Int. Symp. Information Theory*, Yokohama, Japan, 2003, p. 109.

[16] A. Berman and R. J. Plemmons, *Nonnegative Matrices in th Mathematical Sciences*. Philadelphia, PA: SIAM, 1994.

[17] S. Forchhammer and J. Justesen, “Bounds on the capacity of constrained two-dimensional codes,” *IEEE Trans. Inf. Theory*, vol. 46, no. 7, pp. 2659–2666, Nov. 2000.

[18] T. B. Schultz, “Coding for Nanotech Storage,” Master Thesis, Tech. Univ. Denmark, Lyngby, Denmark, 2005.